

A LEVEL

COMPUTER SCIENCE

H446

For first teaching in 2015

Project setting guidance

Version 1

A LEVEL COMPUTER SCIENCE

Introduction	3	Stakeholders during Testing	6
Setting the level for candidates	3	Stakeholders during Evaluation	6
Choosing a suitable project	3	The Iterative process	6
Idea generation	3	Preparation of Candidates	7
Suitable project examples	3	Project idea generation	7
Appraising initial project ideas	3	Indicators of projects with sufficient depth	7
Supporting candidate project choice	3	Persistent data storage	8
Projects with broad scope	4	Programming Paradigms	8
Project with limited scope	4	Interfacing with hardware	8
Project advice from OCR	4	Use of Libraries	8
Suitable Programming Languages	4	Algorithms and Data Structures	8
Additional approved languages	4	Networking and Internet of Things	8
Graphical User Interfaces (GUI)	4	Scope/Depth check	9
Project choice and language selection	5	Indicators of projects with insufficient depth	9
Project and Language combinations	5	MS Access and MS Excel	9
Data Handling	5	Web Based Projects	9
Web-based Data Handling	5	Use of visual programming	9
Mobile Apps/Games	5	Sample project ideas	9
Games	5	Adaptive quiz games	9
Simulations	5	Mobile Phone Apps	10
Robotics	6	Stealth Game	10
Advice on undertaking the Programming Project	6	Scientific Simulations	10
Using stakeholders	6	Scoring Systems	10
Stakeholders and Analysis	6	Candidate Sample Work	11
Stakeholders and Design	6		

Introduction

The A Level Programming Project gives candidates the opportunity to go through the process of developing a substantial piece of software. In order to be able to access the full range of marks available, candidates need to develop a project with enough scope and depth to allow them access to the upper mark bands for each section of the marking criteria.

This guide aims to give an overview of what makes a project suitable. This guide should be used to support and guide project choice, rather than be used as a check list for projects.

In Computer Science, many projects which may lack initial scope can be reviewed and developed to enhance this scope. This guide should help to show where and how early ideas may be developed to create suitable choices for A Level project.

Setting the level for candidates

It is important to appreciate that weaker candidates can still score well by completing projects with a smaller scope.

Stronger candidates will probably naturally select larger projects – and should be encouraged to follow these ideas. It is still possible to obtain the upper mark band with a project that may not fully address the initial scope.

The key to the project is ensuring that the candidate is confident in their ability to solve the problem. Scope may be refined as the project develops.

Choosing a suitable project

Idea generation

Candidates must choose a project individually. Whilst this may initially seem challenging for a candidate, the scope and flexibility of the specification does not limit any particular project ideas.

The key to the project is that it allows candidates to create a **substantial coded element** as part of the solution created.

Suitable project examples

The following are examples of suitable projects. It is not meant to be a definitive list, and we include more ideas later in this guide, but could be used to promote idea generation by candidates:

- ✓ Complex Games (e.g. may involve play against the computer)
- ✓ Simulations
- ✓ Automated scheduling/timetabling
- ✓ Online multi-user websites

The following projects give examples that may limit the scope of the solution and thus the ability to access higher mark bands:

- ✗ Multiple Choice Quizzes
- ✗ Simple data storage and retrieval
- ✗ VBA projects
- ✗ Simple 'single player' games

We would suggest that a candidate tries to create a range of initial ideas for project titles and then reflects back on which seem the most appropriate and engaging for them.

Appraising initial project ideas

Once a candidate has defined some project titles, it is worth asking them to create a short summary of the project. This could include things such as:

- Stakeholders
- Potential research avenues
- Data processing needed
- Current problem/initial ideas for a solution
- Programming Language(s) to be used
- Ideas for a suitable GUI

Once a candidate has created these summaries, they will be better placed to make an informed choice as to which project they wish to engage with.

Supporting candidate project choice

As a teacher, you are able to support your candidates towards choosing suitable ideas for their project. You are allowed to facilitate project choice, but should not give them pre-defined project titles or ideas.

It is important that project choice is driven by the candidate.

Projects with broad scope

It may be that a candidate's project idea has massive scope, in which case you can suggest limiting the scope, and maybe identifying features which could form part of a "wish list".

It is important that candidates appreciate the time constraints that they will have, and that the project does not become cumbersome.

Project with limited scope

Some candidates may struggle with identifying a project that lends itself to decomposition and multiple development cycles. However, many of these ideas may be developed quite easily, which in turn increases the scope of the problem.

Where possible, try to support candidates in developing the idea that they engage with most. This supports their motivation as they develop the project.

Project advice from OCR

OCR does not provide project titles. However, we do support teachers in identifying suitable project scope. OCR will support centres in identifying where project proposals may need more development, or where a candidate's proposal appears to allow full access to the mark bands.

Where a centre wishes to gain feedback, candidate's proposal(s) should be collated into **one single document**.

Each proposal from a candidate should include:

- Title and brief overview of the project
- Programming Language(s) used
- Main project objectives/success criteria
- Why they think this is a suitable project

The file should then be emailed to: ComputerScience@ocr.org.uk

Please note we will only be able to offer guidance on project ideas – and the final choice of project is the centre's responsibility.

Suitable Programming Languages

The focus at A level is for candidates to create a substantial coded solution using a textually-derived high-level programming language.

A list of pre-approved programming languages is included within the Specification in Section 5e.

OCR allows a wide range of languages, some of which are not listed within the specification. Where a candidate wishes to use other languages not listed within the specification or within this document, you must seek confirmation that it will be accepted by OCR **before** commencing the project.

Additional approved languages

The following list of languages have been approved by OCR for use within A Level projects:

- ✓ Swift
- ✓ NodeJS
- ✓ Haskell
- ✓ Unreal/Unity (via C# and C)
- ✓ Lua
- ✓ Robot X
- ✓ Monkey X
- ✓ JavaScript (likely to be used in conjunction with HTML/CSS/MySQL/PHP)

Graphical User Interfaces (GUI)

It is a requirement of the specification that candidates create an appropriate GUI. However, there are **rare** circumstances where this might not be appropriate.

Should a candidate wish to create a program that is more suited to a command line interface then you **must** receive prior authorisation for this from OCR.

Requests should be emailed to: ComputerScience@ocr.org.uk

Project choice and language selection

There is no 'best' language to use for A Level projects. The choice of language(s) used will be dependent on:

- ✓ The project scope
- ✓ Candidate skill level/aptitude
- ✓ Level of familiarity

Some candidates may be confident in attempting a project using a language that may not be taught within the classroom. Some may wish to choose projects that solely use the language they are taught.

Where candidates wish to engage with languages that they are not familiar with – this should be taken into consideration. Learning a new language is fun, but may slow a candidate's progress down.

There is no requirement for all candidates to use the same language for their projects. We welcome a diverse range of projects and language use at A Level.

Project and Language combinations

We offer the following suggestions, linking project ideas to suitable languages.

This is suggested guidance only, and not a definitive requirement.

Data Handling

These projects are relatively simple computationally, and often conceptually similar to traditional ICT database projects. This may be compensated for by a lot of validation checks and sophisticated data structures. There are lots of resources for this family of project, but this type of software is going extinct in this connected world.

Possible languages: VB, Python, C#, C++, C, Objective-C/Cocoa and Delphi (VB and Python seem to be more popular)

Web-based Data Handling

Projects of this style tend to be employment-oriented. Web standards change all the time, and these projects often require knowledge of two languages (e.g. JavaScript

for client-side and PHP/ASP for the server side, plus CSS for the interface). Another consideration is the requirement for paid hosting services or getting the school's IT departments to run a server.

Possible languages: HTML5/JavaScript/PHP, VB/ASP, Java, SQL

Mobile Apps/Games

Whilst these may initially seem easier to do computationally than desktop applications, mobile apps require the knowledge of mobile APIs and server connections.

Candidates will also need to consider testing considerations on the devices that the app/game is designed for.

Possible languages: Objective-C (IOS), Java, Python (via SL4A) for Android, Java for Blackberry, HTML5 for Windows Phones, Django (based on Python), and SWIFT for Apple based OS.

Games

Games tend to be the more popular choice at A Level as they inherently tend to spark creativity. However, a well created and coded game is a significant challenge. We would recommend that any games are focused on quality over quantity.

A well-functioning single level game is likely to lead to higher marks than a massive game which is very simple or has significant flaws. Choice of IDE will be key.

Candidates should be reminded that they are assessed on their coded solution. A focus on graphics and artistry will not solicit credit. Where possible, sourcing free graphics for use is a more efficient use of time!

Possible languages: C#, Python, C++, Objective-C (MacOS), Java, JavaScript, Monkey X

Simulations

Simulations (e.g. Planetary Interactions, Bacteria growth rates, population modelling) have been growing in popularity at A Level, particularly with those studying maths or physics at A Level in conjunction with Computer Science.

Whilst these programs can be designed in a wide range of languages, there are also more focused languages, often specifically designed for this purpose.

Possible languages: MatLab, R, Prolog, Haskell, Python, J

Robotics

Robotics programming is becoming very popular. Robotics programming often includes a level of artificial intelligence with it, and thus can create excellent project choices. Robotics projects may be carried out physically or virtually dependent on the resources available to the school.

There are many environments which allow candidates to program and simulate robotics using high-level textually-derived programs.

Possible languages: C, C++, Python, Robot X

Advice on undertaking the Programming Project

Using stakeholders

The move to use stakeholders, rather than finding a 'real' end user should help speed up the process of project identification. However, effective use of stakeholders to provide feedback and ideas cannot be understated.

For instance, a game which interests the candidate may also interest people of similar age groups. They could then look to expand their stakeholders to include a wider set of demographics.

Stakeholders and Analysis

Stakeholders will be able to guide and provide additional requirements for the project in the analysis stage. Where stakeholders are not used well, it can lead to a very blinkered view of the project. This then reduces the amount of discussion that takes place within the analysis.

If discussion and exploration is limited, justification of success criteria can become shallow. The success criteria may then also not be broad enough and explore the full potential of the project scope.

Stakeholders and Design

Stakeholders can provide valuable interaction during the design process. Stronger projects tend to seek feedback on initial designs and use the feedback mechanism to refine the initial ideas to a final set of designs.

Where interactions are limited, designs can often not anticipate problems which then only become apparent at a later stage which causes issues within the development of the project.

Stakeholders during Testing

Stakeholders will provide valuable feedback, both during developmental testing, and also for the final evaluative testing.

Projects that do not engage with stakeholders for testing can struggle to generate realistic testing feedback and thus fail to highlight usability issues or functionality problems.

Strong interaction with stakeholders during iterative development will increase the chances of creating a successful end product, and provide significant interaction evidence which can then be used within the Evaluations.

Stakeholders during Evaluation

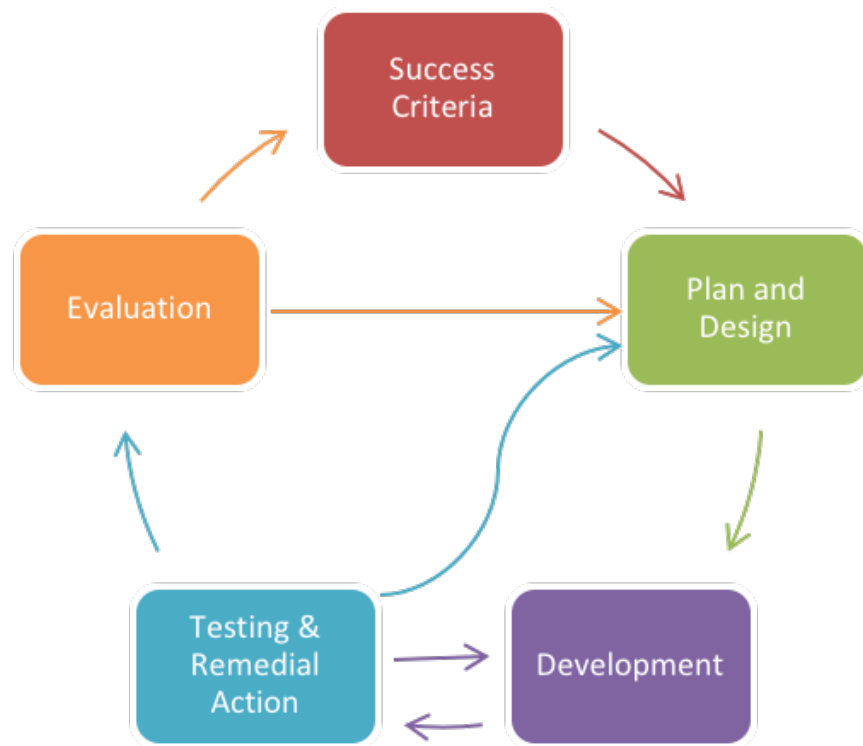
Engagement with stakeholders for evaluation of both iterations and the final product will provide the candidate with much more evidence on which to draw conclusions.

Stakeholders are likely to be able to provide a more critical review of each iteration, or of the final product. This can then be combined with the testing results to build a far more convincing set of evaluations.

The Iterative process

A Level projects should require multiple iterations to create a product from conception to completion.

We provide the following cycle which is likely to take place within the candidate's project.

**Success criteria**

- What will a successful solution do

Planning and design

- Pseudocode, Flowcharts, DFDs, Class Definitions/UML etc.

Development

- Narrative of steps taken
- Annotated Code
- Discussion of methodology

Testing and remedial actions

- Narrative of changes made

Evaluation

- Link to success criteria
- Evidence of success or not

Preparation of Candidates

We would suggest that candidates are familiar with the following before starting the A Level Programming Project.

1. The iterative process for creating their projects
2. The programming techniques listed within the specification
3. Understanding the marking criteria and requirements of the Programming Project
4. Guidance on identifying/selecting a suitable project

For candidates who struggle to identify projects, or for weaker candidates, it is possible to start with a project which has narrower scope, and develop it as progress ensues. A less ambitious project can still score well.

Encouraging candidates to document their development failures and corrections is also encouraged. This will provide much more evidence for justifying the directions candidates chose during their project.

Project idea generation

One way you can stimulate ideas and discussion around projects ideas is to use the following method:

- Have a 'brainstorm' session with the group to create generic ideas in conjunction with this guidance document and the specification
- Ask each candidate to then use these ideas to create a list of 10 to 15 projects
- The candidate then selects their Top 3 preferred ideas
- Develop a short proposal for each of those three ideas
- Discuss the final project proposal within the group
- Have a vote across the class on each candidate's Top 3 suitability
- Pick the one they feel most confident in and submit to OCR for review.

Indicators of projects with sufficient scope

The following list is not meant to be a checklist of requirements. If a project includes one or more of these indicators, then it is likely to be able to provide suitable scope for an A Level project. The more indicators that exist – the stronger the likelihood it would be suitable.

Persistent data storage

Most programs will need some form of persistent storage.

This may be as simple as storing user preferences or may be as complex as storing large amounts of data to be analysed and generate reports. Candidates will be expected to choose suitable methods of storing this data.

In some cases, a simple text file or CSV is sufficient whereas in others it may be appropriate to use storage methodologies such as JSON, XML or SQL. More complex data storage can be indicative of a project with more scope and depth.

Programming Paradigms

There is no set paradigm for the project. Candidates may choose a procedural approach but others are equally acceptable. Some projects may be suited to object-oriented, functional or logical programming paradigms.

Whilst by no means a requirement, a candidate choosing an object-oriented paradigm is a good indication that the project has good scope.

Where the project has scope for the candidate to demonstrate programming techniques or algorithms from the A Level Specification in their chosen methodology, this too is an indication of a good level of depth and scope. They may also use techniques specific to the language.

For example, a candidate programming in an object-oriented language may structure their program using a model-view-controller pattern, a candidate creating a web-based project may use AJAX for improved responsiveness.

Interfacing with hardware

Some candidates may choose to interface with particular hardware such as sensors, or with features on mobile devices, such as QR scanners or cameras.

Projects that do this are likely to need to access libraries/APIs which is a good indicator that the project has sufficient scope.

Use of Libraries

Libraries allow candidates to add functionality to their projects that they may not have the time or expertise to code themselves.

Candidates will not be penalised for the use of libraries as use of libraries to tackle parts of their problem is best practice. Candidates must still ensure that there is a significant self-coded element to their project, and this is usually achieved through the code needed to interact with these libraries within their project.

Sensible choice and use of libraries can be an indicator of a good level of scope in a project.

Candidates must be reminded to clearly reference where libraries are used.

Algorithms and Data Structures

There is no requirement for candidates to reinvent the wheel! If a program requires sorting of a data structure, then a candidate deserves no less credit for using a language's built-in sort routine (as a professional programmer likely would) than if they code their own sorting algorithm.

Similarly, if a required data structure is offered by a language a candidate would be well advised to use them, rather than recreating them.

This said there will be the need for algorithms that aren't available natively. These might be established algorithms such as Dijkstra's Shortest Path or MinMax which the candidate needs to implement or an algorithm the candidate has had to devise for themselves.

Likewise, existing data structures may need to be adapted to best suit the project.

A project where a candidate shows clear thought into developing their data structures and algorithms and using them alongside or integrating them with pre-existing data structures and algorithms is indicative of good project scope.

Using the algorithms contained within the specification are a good starting point for project ideas, were appropriate.

Networking and Internet of Things

Projects can often have their scope broadened through the use of interaction across a network.

For instance, a simple two-player game can be developed by allowing network play. Similarly, there are many good projects that can be developed using the Internet of Things (IoT) and beyond.

Combination and utilisation of differing technologies will often provide great scope and ideas for a programming project, and be engaging due to the potential for real life use. For instance, creating an IoT project that affects the candidate's day-to-day life.

Scope/Depth check

- ✓ Permanent Data Storage
- ✓ Potential for OO Paradigm
- ✓ Interfacing with hardware/Networks
- ✓ Exporting/Linking with other software
- ✓ Programs that learn/adapt over time
- ✓ Games/Physics
- ✓ Extended Logic Chains
- ✓ Use of libraries
- ✓ Use of A Level sorting/searching techniques
- ✓ Combination of differing technologies
- ✓ Expert Systems
- ✓ Simulators

Indicators of projects with insufficient scope

This list is not exhaustive, and highlight the most common pitfalls we see when discussing suitable project proposals from candidates with centres.

MS Access and MS Excel

Using applications such as MS Access or MS Excel and adding scripting will not offer the depth necessary for an A Level project as they can be overly restrictive and do not give the candidate the opportunity to demonstrate the required decision-making skills.

Often too much of the functionality that would be expected to be developed by the candidate is built into these applications.

Candidates are expected to produce a stand-alone solution; this is not possible when using these applications.

Web-Based Projects

Web-based projects have the potential to have more than enough scope for A Level but caution needs to be taken. It is easy for web-based projects to lack depth and scope.

Static HTML sites will limit a candidate's access to the full range of marks.

Just because a site is dynamic that does not in itself necessarily allow a candidate to show the skills necessary for upper mark band access.

A site which just has a login page and simply retrieves and displays content from a database is not in itself sufficient.

Use of visual programming

Visual drag and drop languages such as Scratch, AppInventor and BYOB are not suitable to produce a program for an A Level project.

Any IDE which allows pre-generation of the majority of code is not likely to allow candidates to show enough individuality in code creation. We do not authorise the use of IDEs where this is the case. If you are in doubt as to the suitability of an IDE, please do contact us.

Sample project ideas

These ideas are not meant to form templates for projects – and candidates should not just be given a list to choose from. They do however provide stimulus for discussion and give examples of the level of project scope we look for at A Level.

Adaptive quiz games

A candidate decides to make a revision program for candidates. It will store the scores each candidate gets when taking a test. The system then adapts the questions it asks according to previous performance and subject content, graphically showing the teacher how the candidate progresses over time.

This style of project has wide scope to demonstrate appropriate skills at A Level.

Key parts to this project could include:

- Storing the candidate's performance in a relational database
- Devising an algorithm to continually learn which questions the candidate needs more help with
- Using a library to generate graphs for the teacher
- Create suggestions/reports as to revision areas for the candidate

Mobile Phone Apps

Many candidates choose to make apps. One example could be a local council requiring an app to increase awareness of mental health issues.

The app allows the user to access information regarding mental illness and allows them to take quizzes to test their awareness of mental health issues. The app also needs to be useable when the user's device was 'offline'. Once internet connection is established again, the app must sync with the remote database.

The app contains a content management system so the council could update the information and quizzes when they wanted. Due to its nature, the app communicates with a central database to keep itself 'updated' adding synchronisation and networking into the scope of the project.

Stealth Game

A candidate has a particular enthusiasm for stealth-based games. Using peers as a user group they act as their own user and create a game which involves sneaking past guards. Levels became more complex. The aim of the game is to recover different objects each level. Each of the guards move around using artificial intelligence and pathing. Each guard's sight is affected by things such as lighting and objects being in the way.

The candidate investigates and adapts existing algorithms to simulate the guards' field of vision. This involves a significant amount of research and prototyping.

The game was coded in C# using a game engine. The game uses complex libraries and learning algorithms.

Scientific Simulations

A candidate creates a program for a company. One of the tasks the company does is analysing temperature data from boreholes to determine if they have reached the bottom of the borehole. This is done through a complex set of calculations called Horner correction.

The candidate wants to write an application to perform this analysis.

The candidate proposes to adhere to the company's software engineering guidelines whilst doing so.

The project is developed using the model-view-controller pattern and set up in such a way that it could be analysed using the company's automated testing software.

Whilst the front end of the system is relatively simple there is a great deal of depth present. The project itself will be coded using an object-oriented approach following an MVC (Model-View-Controller) pattern.

The calculations require use of matrices and statistical calculations. The proposal is to integrate the functionality of specialist mathematical libraries into the coding of the Horner Correction algorithm. A graph library will be used to help produce easily readable reports.

Scoring Systems

A school wishes to develop a Sports Day recording program.

Records are kept of each year group and event, and trophies awarded to the most successful of a school's four 'Houses'.

In previous years they had used a set of spreadsheets to keep track of the scores but this was error prone and required a lot of effort to be duplicated.

This school wants a web-based score keeping system. The system is proposed to be coded in PHP and will run off a computer acting as a webserver on the LAN.

Multiple users will be able to update the scoring system as results for events were coming in. The system will instantly show whether any school records had been broken and how each of the school's 'Houses' were performing.

The Sports Day organiser will be able to log onto the system on his tablet and access automatically updated results.

The results are planned to be stored in a relational database. The front end will use jQuery libraries in conjunction with JavaScript code. AJAX will be used to help improve the responsiveness of the application.

The use of synchronous updating and multi-user remote access adds sufficient depth to this project.

Candidate Sample Work

OCR produces Candidate Sample Work to help centres in both identifying and assessing A Level projects.

The Candidate Sample Work covers a range of projects from High to Low grades, giving ideas of how projects have been developed.

Links to Candidate Sample Work (Candidate Exemplars) may be found on the H446 webpage, under the [Assessment tab](#).

Please note that Interchange access is required to access Candidate Sample Work.



We'd like to know your view on the resources we produce. By clicking on the 'Like' or 'Dislike' button you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click 'Send'. Thank you.

Whether you already offer OCR qualifications, are new to OCR, or are considering switching from your current provider/awarding organisation, you can request more information by completing the Expression of Interest form which can be found here:

www.ocr.org.uk/expression-of-interest

OCR Resources: *the small print*

OCR's resources are provided to support the delivery of OCR qualifications, but in no way constitute an endorsed teaching method that is required by OCR. Whilst every effort is made to ensure the accuracy of the content, OCR cannot be held responsible for any errors or omissions within these resources. We update our resources on a regular basis, so please check the OCR website to ensure you have the most up to date version.

This resource may be freely copied and distributed, as long as the OCR logo and this small print remain intact and OCR is acknowledged as the originator of this work.

Our documents are updated over time. Whilst every effort is made to check all documents, there may be contradictions between published support and the specification, therefore please use the information on the latest specification at all times. Where changes are made to specifications these will be indicated within the document, there will be a new version number indicated, and a summary of the changes. If you do notice a discrepancy between the specification and a resource please contact us at: resources.feedback@ocr.org.uk.

OCR acknowledges the use of the following content:
Square down and Square up: alexwhite/Shutterstock.com

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: resources.feedback@ocr.org.uk

Looking for a resource?

There is now a quick and easy search tool to help find **free** resources for your qualification:

www.ocr.org.uk/i-want-to/find-resources/

www.ocr.org.uk

OCR Customer Contact Centre

General qualifications

Telephone 01223 553998

Facsimile 01223 552627

Email general.qualifications@ocr.org.uk

OCR is part of Cambridge Assessment, a department of the University of Cambridge. For staff training purposes and as part of our quality assurance programme your call may be recorded or monitored.

© **OCR 2018** Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee. Registered in England. Registered office The Triangle Building, Shaftesbury Road, Cambridge, CB2 8EA. Registered company number 3484466. OCR is an exempt charity.



**Cambridge
Assessment**

